# SPsec102 – Glossary

This document summarizes the terms used by the SPsec (Small-Packet Network Security Sublayer) documents. Grouped by functionality, not alphabetically.


The SPsec specifications are divided into the following documents:

- **SPsec101** – Small-Packet Network Security Concept
  Introduction to the security concept of SPsec.
- **SPsec102** – Small-Packet Network Security Glossary
  Terminology and references used by SPsec.
- **SPsec201** – Small-Packet Network Security Generic Specification
  Network independent specification of the SPsec technology and methods.
- **SPsec301** – Small-Packet Network Security Serial Channel Mapping
  Mapping SPsec to generic serial point to point communication.
- **SPsec302** – Small-Packet Network Security CAN FD Mapping
  Mapping SPsec to CAN FD, supporting CANopen FD and J1939 FD.

Version 1.26 of 11-DECEMBER-2025, jointly authored by


www.em-sa.com
Embedded Systems Academy GmbH
Bahnhofstraße 17
30890 Barsinghausen, Germany


ivesk.hs-offenburg.de
Hochschule Offenburg
Badstraße 24
77652 Offenburg, Germany

This project has been funded as part of the Central Innovation Program for SMEs (ZIM) by the Federal Ministry for Economic Affairs and Climate Action (BMWK).

# Contents

# 1 SPsec Basics

These terms provide the basic definitions used by SPsec.

**sub-layer security:** the security provided by SPsec is implemented as a sub-layer to a network layer. Default placement of SPsec is in the layer directly above the Data Link Layer. Typically this is the Network Layer, but not all small-packet networks have that implemented.

**multi-participant security:** SPsec security does not only provide “1:1” but also “N:M” (many to many) security. Multiple devices are combined in a secure group session and can exchange data units securely based on the same shared communication key. The group authentication is limited to “the data unit comes from within the secure group”, without providing a device granular authentication.

**small-packet network:** network system suitable for implementing SPsec. Common characteristics include packet sizes of just a few to a few hundreds of bytes, bitrates and number of data units exchanged are limited to the point where also lower performance microcontrollers can handle them. Use candidates are I2C, LIN, Modbus RTU, CAN, CAN FD and similar technologies.

**participant id, node id:** all nodes in a network are addressable using a uniquely assigned node id. If the network does not offer this functionality, it must be added by the mapping document.

**grey channel:** in secure communication, a white channel is a communication channel considered secure, a black channel is considered unsecure (we do not know if and which security relevant measures are available). Small-packet networks are typically somewhere in between as there can be physical access limitations, injection detection or other non-cryptographic security measures in place.

**commander (previously master) and responder (previously slave):** some small-packet networks use a single commander to initiate all communication. The responders only start transmission after they have been addressed by the commander.

**single-access arbitration:** bus access is controlled by a single commander that addresses the responders individually.

**multi-access arbitration:** all nodes can transmit their addressed data units at any time. Collisions are resolved by hardware.

**addressed data unit:** communication packet as used by the small-packet network protected by SPsec, contains data and meta data (typically an address value involving a node-id).

**secure addressed data unit:** a protected addressed data unit with an added security stamp and optionally the data encrypted.

**security stamp:** data added to an addressed data unit to protect it. Contains an authentication tag and additional meta data required to handle SPsec communication.

**secure SPsec session:** an optimized session-based protocol optimized for small- packet networks. It provides both point to point authentication and common key derivation. Used for the initial

authentication of communication partners and for configuration. Loosely based on TLS-PSK (Transport Layer Security with Pre-Shared Keys).

**secure group:** a set of devices configured to communicate with each other using shared security credentials. Also see multi-participant security.

**SPsec security level 1: group authentication with post-event assurance:** at this security level all addressed data units exchanged are continuously monitored for security events. Once a security event like an injection is detected, it is reliably reported, for example by stopping secure heartbeat production. Security events are therefore reported after the fact. The exact delay depends on timeout settings.
NOTE: not all small packet networks are suitable for this method.

**SPsec security level 2: group authentication with instant-event assurance:** at this security level authentication is added to SELECTED addressed data units. Security events can be directly detected by the receiver(s) of the addressed data units before its content is used. May use lightweight cryptographic methods.

**SPsec security level 3: group security with instant-event assurance:** at this security level both authentication and encryption are provided for ALL addressed data units. Security events can be directly detected by the receiver(s) of the addressed data units before its content is used.

**internal control plane:** communication between host application and SPsec roles within a device. The network's addressed data units are used for communication between the security sub-layer and the application. That requires reserving at least one of the available addresses for this purpose.

**external control plane:** communication between SPsec roles of all participating devices. The network's (secured) addressed data units are used for communication between the security sub-layers of the participants. That requires reserving some of the used addresses for this purpose.

**data plane:** communication between host applications running on connected devices. When passing through the SPsec, there is a transition from addressed data units to secure addressed data units and back.

**mapping document:** the SPsec documents defining how SPsec is mapped to a specific small-packet network technology.

**original equipment manufacturer (OEM):** in SPsec this is the party responsible for network security over the entire lifetime of the product embedding the small-packet network. Includes responsibility for the secure installation of the integration keys (and optional provisioning key). Additional responsibility not part of SPsec definitions includes security protection at higher layers and the entire apparatus using the network.

**repeater, bridges & gateways:** as is, SPsec is not intended to work across repeaters, bridges or gateways. A detailed review of all mechanisms is required, when security is required across repeaters, bridges and gateways.

# 2 Attack Vectors of Small-Packet Networks

The following attack vectors are typical threats for small-packet networks and can all be addressed by SPsec.

**sniffing (attack):** an attack where addressed data units are read by an attacker for further analysis. Could be used to access personal data or draw conclusions about intellectual property.

**replay (attack):** an attack where previously recorded (sniffed) sequences of addressed data units are injected into the network again with the purpose of triggering functionalities that require a certain sequence of commands. Could be used to unlock sealed doors or compartments or force the equipment to perform actions desired by the attacker.

**spoofing (attack):** an attack where addressed data units get injected into the network to get receivers to accept data or commands that are not coming from the original source. Could be used to manipulate data to exceed safety limits or change fiscal data.

**counterfeit parts (infiltration, masquerade):** unauthorized use of fake or substandard components in place of genuine parts in a system. Such counterfeit parts can compromise system performance, reliability, and safety.

**re-purposing:** in SPsec this is the threat that individual network devices get stolen and can easily be re-used in other systems.

# 3 SPsec Cryptographic Methods and Inputs

Which specific cryptographic methods are used is defined in the mapping document applying SPsec to a specific network technology. The following methods and inputs are required.

**authentication tag:** a message authentication code that ensures the authenticity and integrity of secured communication and covers the entire addressed data unit. The authentication tag is generated and checked using an AEAD interface.

**authentication method:** cryptographic method chosen within the AEAD implementation to generate a message authentication code suitable to check the authenticity and integrity of the data. Inputs are the entire addressed data unit (data and meta data) to authenticate, the uniqueness value (timestamp or counter) and a key. Output size is specified by the mapping document.

**encryption method:** cryptographic method chosen within the AEAD implementation to encrypt the data in the addressed data unit. Inputs are the data and a key. Output must be of the same size as the input.

**authenticated encryption with associated data (AEAD):** a single shot primitive that (optionally) encrypts plaintext and authenticates additional associated data using a nonce and a key.

**secure heartbeat:** for systems with multi-access arbitration this is periodically produced by all participants in secure state and recommended to be consumed by all participants. Uses an

authentication tag based on AEAD and a combination of uniqueness and salt with the communication key.

**secure node guarding:** for single commander systems the commander periodically polls this security status information from all participants. Uses an authentication tag based on AEAD and a combination of uniqueness and salt with the communication or session key.

**key derivation function (KDF):** cryptographic method chosen to derive a new key based on another key (higher in the key hierarchy). Inputs are a nonce or a salt and the key used as a base.

**uniqueness:** for protection against replay attacks and to avoid nonce reuse, secure data exchanged requires a uniqueness value. This is added as an input to the authentication method. Typically, a timestamp or transmit counter value.

**nonce:** number used once. A random (or counter based) value added to an input of cryptographic functions to provide essential uniqueness. If counters or timestamps are used, measures must be taken to ensure that the same nonce is not used with the same key more than once.

**nonce misuse:** occurs when a nonce is reused with the same key, which can critically compromise AEAD-based encryption schemes such as AES-GCM. By building nonces based on a high-resolution timestamp and node id of the sender it can be ensured that there is no re-use.

**salt:** a random value added to an input of cryptographic functions to provide some additional uniqueness. Typically, not as essential as a nonce, provides more and changing data (like giving random values to padding bytes).

**pre-shared salt:** for each key there is also a pre-shared salt value. If for any cryptographic function using salts, the salt available is shorter as required, then it is extended with data from the pre-shared salt.

**timestamp:** if a timestamp is used as uniqueness value, then all participants must maintain a synchronized timestamp including date information. Each transmission must use a different timestamp value to ensure uniqueness. Resolution is defined by the mapping document.

**(transmit) counter:** if a counter is used as uniqueness value, then both transmitter and all receivers must maintain such an individual counter for every address of the secure addressed data units communicated.

# 4 SPsec Roles

The tasks performed to operate SPsec are divided into “roles”.
NOTE: configurator and timestamp sync role may be combined.

**participant role:** instance capable of participating in the secure communication. Mandatory for all devices participating in the secure communication.

**participant-id:** the node-id of a participant.

**(time) sync role:** instance responsible for synchronizing the uniqueness value. If timestamps are used, provides the timestamp to be synchronized to. If transmit counters are used, provides the initial value of a transmit counter. Must implement a participant role to participate in secure communication.

**configurator role:** instance responsible for the configuration of the secure group(s). Must implement a participant role to participate in secure communication. Does not need to be present during regular operation.

# 5 SPsec Participant States (Finite State Automaton)

Each participant implements an FSA with the following three states. All state transitions are handled by the Participant role itself.

**waiting:** participants in this state wait for the authenticated sync value to be received.

**secure:** participants in this state actively participate in the secure communication.

**warning:** participants in this state detected a security warning, like a secure heartbeat timeout or an authentication failure. Depending on configuration and severity of the warning, Participants may choose to resume (go back to “secure” state) or abort (got to “waiting” state).

**configuration:** participants in this state are in an active secure configuration session.

# 6 SPsec keys

SPsec uses a symmetric cryptographic key (known only to legitimate parties) as input to cryptographic methods. Key length is defined by the SPsec mapping documents.

**communication key:** current key used to secure the communication, cyclically derived from the seed key.

**seed key:** key used as a base for deriving the communication keys. Should be updated or regenerated with every maintenance cycle.

**integrator key:** the system integrator’s root of trust. This can be unique per device but for simplicity an integrator may choose to share this key among all devices of one network. Typically installed by OEM when powering up the system for the first time.

**provisioning key:** optional device specific key preinstalled by the device manufacturer. This key is only used to protect the transfer of the integrator key to one or multiple devices.

**key selector:** an unsigned value selecting one of the keys above.

**key id:** each key is associated (by the configurator) with an id. A configurator not knowing which keys are present in a device may read this information even if a secure configuration session was not yet established.

**key hierarchy:** the defined keys build a hierarchy. From lowest to highest trust: communication key < session key < integrator key < provisioning key. If a key is not available or fails, the system can fall back to the next higher priority key.

**key rotation:** time or counter based repetitive cycle after which a new communication key is derived from the seed key. Details to be defined in the network mapping document.

**key storage:** the integrator and seed key need to be stored in (preferably secure) non-volatile memory by all participants. If SPsec has access to (secure) NVOL memory, it shall store and retrieve it directly. Otherwise, it is passed via the internal control plane to the app, which needs to store and retrieve it. In this case it is recommended for SPsec to encrypt/decrypt the key before passing it on via the control plane.

**key management:** in SPsec, the communication key is managed internally within the security sub-layer. The integration and seed keys (should be unique for each network system) must be installed on all devices that should participate in the secure group. This typically happens when system is powered up the first time and during maintenance.

**initial key provisioning:** if participants have the capability of using public-key certificates, then those can be used to mutually authenticate and to protect the exchange of the integrator key. Alternatively, if devices have a pre-installed provisioning key, then this can be used to protect the exchange of the integration key on first start-up or during maintenance. Last, if production and maintenance (add-on of devices) always happens in a secure environment, then no protection might be required, and the integration key can be distributed without protection.

# 7 SPsec Data Objects

These data objects are used by SPsec. When used on the external control plane, all communication objects must be secured using SPsec methods. Content details are defined in the mapping documents.

**sync communication object:** data object transmitted by the sync role to implement timestamp or transmit counter synchronization.

**configurator communication object:** data object transmitted by the configurator role to configure participants.

**participant communication object:** data object transmitted by the participant role as a response to configurator requests or to indicate an event like a security warning.

**security stamp object:** data object appended to an addressed data unit to form a secure addressed data unit.

# 8 SPsec Times and Timeouts

These times and timeouts are used by SPsec. Their default values are defined in the mapping documents.

**session response timeout:** timeout value used by a client when waiting for a server response.

**session overall timeout:** timeout used by a server to detect client inactivity.

**timestamp acceptance window:** when timestamps are used, any participant receiving a secure addressed data unit check, if the timestamp in its security stamp is within a plus / minus range of the participant's own synchronized timer. If it is outside the acceptance window the addressed data unit is not accepted.

**timestamp synchronization cycle time:** when timestamps are used, cycle time used by the timestamp synchronization role to transmit the timestamp communication object.

**timestamp synchronization expiration timeout:** when timestamps are used, the timeout after which timestamp synchronization is considered expired, causing participants to enter the warning state. This timeout is reset with each timestamp synchronization.

**secure heartbeat time:** cycle time used by all participants to produce the secure heartbeat.

**secure heartbeat timeout:** when consuming secure heartbeats of a participant, that participant is considered disconnected or failed when this time passes without receiving any secure heartbeat from that participant.

**communication key rolling time:** cyclic time used based on the synchronized time to derive a new communication key.

**warning state hold timeout:** after entering the warning state, participants start this timeout. If no new security warning occur within this timeout, they return into the "secure" state.